

AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions of claims in the application:

Listing of Claims:

1. (Currently amended) Computer executable instructions for performing [[A]] a method of expressing compiling an intermediate language representation of a source language computer program and providing the intermediate language representation to a target runtime wherein an association between a declaration and an implementation are expressed in a language neutral fashion, the computer executable instructions stored on computer readable media, the method comprising:

determining whether a source language association rule related to a the declaration is different from a default association rule for the target runtime;

~~expressing compiling~~ an association between the declaration and ~~an the~~ implementation ~~as into~~ [[an]]its intermediate language representation according to an override association rule for the target runtime if the source language association rule is different from the default association rule for the target runtime;

~~expressing compiling~~ an association between the declaration and the implementation ~~as into~~ [[an]]its intermediate language representation according to the default association rule if the source language association rule is the same as the default association rule for the target runtime; and

providing the intermediate language representation of the association between the declaration and the implementation to the target runtime to compile the intermediate language representation into native code.

2. (Currently amended) The method of claim 1, wherein ~~expressing compiling~~ the association according to the override association rule comprises expressing an explicit association between the declaration and the implementation.

3. (Original) The method of claim 2, wherein expressing the explicit association comprises creating an association between a class, a code body associated with the implementation, and the declaration.
4. (Previously presented) The method of claim 3, wherein creating the association between the class, the code body associated with the implementation, and the declaration comprises creating an entry in an override association table having entries for the class, the code body, and the declaration.
5. (Currently amended) The method of claim 1, wherein the declaration comprises a method declaration signature, the implementation comprises an implementation signature, the default association rule for [[a]]the target runtime comprises signature matching, and wherein ~~expressing~~ compiling the association between the declaration and the implementation according to the override association rule for the target runtime if the source language association rule is different from the default association rule for the target runtime comprises expressing the association between the method declaration signature and the implementation signature if the source language association rule is different from signature matching.
6. (Currently amended) The method of claim 5, wherein ~~expressing~~ compiling the association comprises creating the association between a class, a code body associated with the implementation signature, and the method declaration signature.
7. (Previously presented) The method of claim 6, wherein creating the association between the class, the code body associated with the implementation signature, and the method declaration signature comprises creating an entry in an override association table having entries for the class, the code body associated with the implementation signature, and the method declaration signature.

8. (Previously presented) The method of claim 5, wherein the method declaration signature matches a first implementation signature associated with a first implementation in the class and a second implementation signature associated with a second implementation in the class, and wherein expressing the association between the method declaration signature and the implementation signature according comprises expressing an explicit association between the method declaration signature and one of the first and second implementation signatures.

9. (Previously presented) The method of claim 8, wherein expressing the explicit association comprises creating the association between the class, a code body associated with one of the first and second implementations, and the method declaration signature.

10. (Previously presented) The method of claim 9, wherein creating the association between the class, the code body associated with one of the first and second implementations, and the method declaration signature comprises creating an entry in an override association table having entries for the class, the code body associated with one of the first and second implementations, and the method declaration signature.

11. (Previously presented) The method of claim 8, further comprising selecting one of the first and second implementations for association with the method declaration signature according to the source language association rule.

12-27. (Canceled).

28. (Currently amended) A computer-readable medium having computer-executable instructions for:

determining whether a source language association rule related to a declaration is different from a default association rule for a target runtime;

~~expressing compiling~~ an association between the declaration and an implementation ~~according to an override association rule for the target runtime as an into~~ intermediate language code ~~according to an override association rule for the target runtime~~ if the source language association rule is different from the default association rule for the target runtime;

~~expressing compiling~~ an association between the declaration and the implementation ~~according to the default association rule as an into~~ intermediate language code according to the default association rule if the source language association rule is the same as the default association rule for the target runtime; and

providing the intermediate language code of the association between the declaration and the implementation to the target runtime to compile the intermediate language code into native code.

29. (Currently amended) The computer-readable medium of claim 28, wherein the computer-executable instructions for ~~expressing compiling~~ the association according to the override association rule comprises computer-executable instructions for expressing an explicit association between the declaration and the implementation.

30. (Previously presented) The computer-readable medium of claim 29, wherein the computer-executable instructions for expressing the explicit association comprises computer-executable instructions for creating the association between a class, a code body associated with the implementation, and the declaration.

31. (Previously presented) The computer-readable medium of claim 30, wherein the computer-executable instructions for creating the association between the class, the code body associated with the implementation, and the declaration comprises computer-executable instructions for creating an entry in an override association table having entries for the class, the code body, and the declaration.

32. (Previously presented) The computer-readable medium of claim 28, wherein the declaration comprises a method declaration signature, the implementation comprises an implementation signature, the default association rule for the target runtime comprises signature matching, and wherein expressing an association between the declaration and the implementation according to the override association rule for the target runtime if the source language association rule is different from the default association rule for the target runtime comprises expressing the association between the method declaration signature and the implementation signature if the source language association rule is different from signature matching.

33. (Previously presented) The computer-readable medium of claim 32, wherein the method declaration signature matches a first implementation signature associated with a first implementation in a class and a second implementation signature associated with a second implementation in the class, and wherein the computer-executable instructions for expressing the association between the method declaration signature and the implementation signature if the source language association rule is different from signature matching comprises computer-executable instructions for expressing an explicit association between the method declaration signature and one of the first and second implementation signatures.

34-40. (Canceled).

41. (Currently amended) A computer-implemented system for expressing compiling an association between a declaration and an implementation ~~in a language neutral fashion, the computer-implemented system stored on computer-readable media, the computer-implemented system~~ comprising:

a first component to determine whether a source language association rule related to the declaration is different from a default association rule for a target runtime; and

an association expression component to express compile the association between the declaration and the implementation ~~according to an override association rule for the target runtime as into~~ into ~~[[an]]its~~ intermediate language representation ~~according to an override association rule for the target runtime~~ if the source language association rule is different from the default association rule for the target runtime, ~~and to express compile~~ the association between the declaration and the implementation ~~according to the default association rule as into~~ into ~~[[an]]its~~ intermediate language representation ~~according to the default association rule~~ if the source language association rule is the same as the default association rule for the target runtime, ~~wherein and to provide the intermediate language representation is provided of the association between the declaration and the implementation to the target runtime to compile into machine executable code.~~

42. (Currently amended) The computer-implemented system of claim 41, wherein the association expression component expresses an explicit association by creating the association between a class, a code body associated with the implementation, and the declaration.

43. (Currently amended) The computer-implemented system of claim 42, wherein the association expression component creates an entry in an override association table having entries for the class, the code body, and the declaration.

44-46. (Canceled).

47. (Currently amended) A source compiler embodied on computer-readable medium for generating an intermediate language representation of a source code program, comprising:

an association expression system for expressing an association between a source language declaration and an implementation in a language neutral fashion, the association expression system comprising:

a first component to determine whether a source language association rule related to the source language declaration is different from a default association rule for a target runtime; and

an association expression component to ~~express~~ compile the association between the source language declaration and the implementation ~~according to an override association rule for the target runtime as into~~ into ~~[[an]]~~ intermediate language code according to an override association rule for the target runtime if the source language association rule is different from the default association rule for the target runtime, the association expression component further to ~~express~~ compile the association between the declaration and the implementation ~~according to the default association rule as into~~ into ~~[[an]]~~ intermediate language code according to the default association rule if the source language association rule is the same as the default association rule for the target runtime, ~~and to provide wherein the intermediate language code is provided of the association between the declaration and the implementation to the target runtime to compile into machine executable code.~~

48. (Canceled).